UART喚醒並且頻率補償

NuMicro® 32 位系列微控制器範例代碼介紹

文件資訊

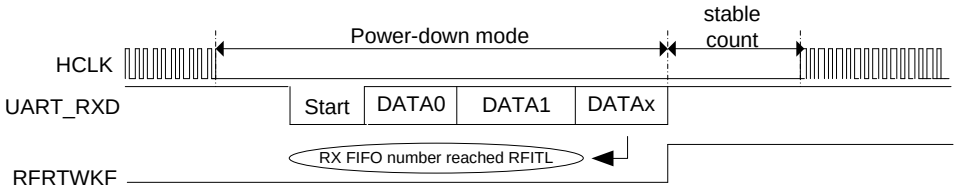
代碼簡述	本範例代碼使用UART把M480從低功耗模式喚醒並進行頻率補償
BSP 版本	M480 Series BSP CMSIS V3.04.000
開發平台	NuMaker-PFM-M487 Ver 3.0

1 功能介紹

1.1 簡介

此範例程式中，M480會進入power down模式，當所收到UART的資料達到門檻值，即會喚醒M480。由於power down模式下 UART 時鐘源需要設為 LXT，因此使用UART鮑率補償以達到更準確的頻率。使能暫存器WKRFRTEN(UART_WKCTL[2])可開啟UART喚醒功能。

進入power down模式後，當RX FIFO裡接收到的資料達到了門檻值設定 RFITL (UART_FIFO[7:4])，會觸發接收資料標誌 RFRTWKF(UART_WKSTS[2])。下圖是UART接收資料達到門檻值喚醒示意圖：



1.2 原理

使用者可以在BRCOMP(UART_BRCOMP[8:0])設定需要補償的位元，並且在BRCOMPDEC(UART_BRCOMP[31])設定正補償或是負補償，使所設定的位元資料時間增加或減少1個UART的時脈。以下為暫存器說明：

Bits	Description	
[31]	**BRCOMPDEC**	**Baud Rate Compensation Decrease** 0 = Positive (increase one module clock) compensation for each compensated bit. 1 = Negative (decrease one module clock) compensation for each compensated bit.
[30:9]	**Reserved**	Reserved.
[8:0]	**BRCOMP**	**Baud Rate Compensation Patten** These 9-bits are used to define the relative bit is compensated or not. BRCOMP[7:0] is used to define the compensation of UART_DAT[7:0] and BRCOM[8] is used to define the parity bit.

範例如下：

UART peripheral clock = 32.768K (LXT)

UART Baud rate = 9600, 1bit = 104.167us

原波形會產生3個UART時脈的時間來表示1個位元的UART資料 ((1 / 32.768K) * 3 = 91.55us)，波形如下：

設定補償暫存器如下：

BRCOMP[8:0] = 0x0010100101

BRCOMPDEC[31] = 0

因此在bit0, bit2, bit5, bit7會補償1個UART時脈的時間 = ((1 / 32.768K) * (3+1) = 122.07us)

波形如下：

開啟補償功能可以控制UART鮑率的誤差在± 0.5個UART時脈的時間以內。

2 代碼介紹

首先設定UART1的傳輸速率為9600，並且對bit0, bit2, bit5, bit7做頻率補償。接著開啟UART喚醒功能，再來設定UART接收資料的門閥值。其程式碼如下：

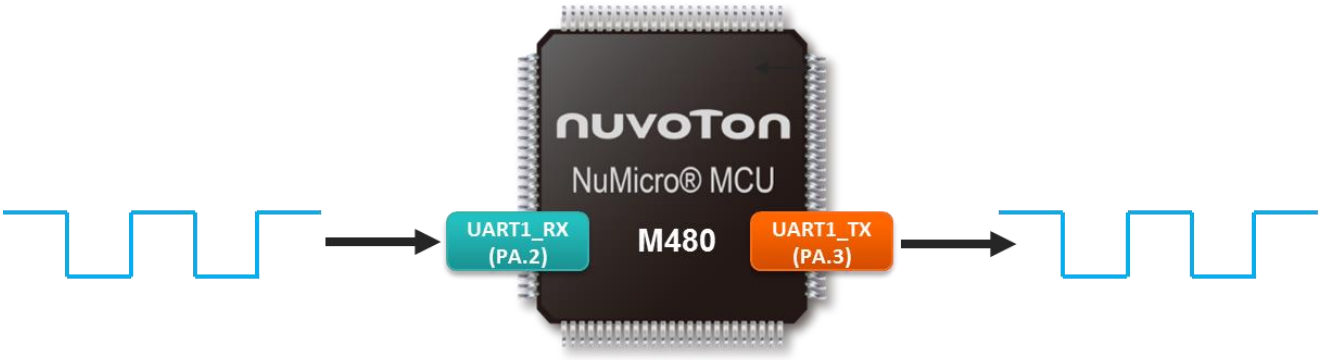
```
void UART_FunctionTest()
{
    SYS_UnlockReg();

    /* Configure UART1 and set UART1 baud rate */
    UART_Open(UART1, 9600);

    /* Set Compensation in bit0, bit2, bit5, bit7 */
    UART1->BRCOMP = 0xA5;

    g_bWait = 0;

    UART1->WKCTL = UART_WKCTL_WKRFRTEN_Msk;

    /* Enable UART RDA/THRE/Time-out interrupt */
    NVIC_EnableIRQ(UART1_IRQn);
    UART_EnableInt(UART1, (UART_INTEN_WKIEN_Msk));

    CLK->PWRCTL |= CLK_PWRCTL_PDWKIEN_Msk;
    NVIC_EnableIRQ(PWRWU_IRQn);

    /* clear status */
    UART1->WKSTS = UART1->WKSTS;

    /* Wait debug message finish */
    while((UART0->FIFOSTS & UART_FIFOSTS_TXEMPTYF_Msk) == 0);

    UART1->FIFO &=~ UART_FIFO_RFITL_Msk;
    UART1->FIFO |= UART_FIFO_RFITL_4BYTES;

    /* Enter to Power-down mode */
    CLK_PowerDown();

    while(!g_bWait);
    CLK_SysTickLongDelay(1000000);
```

```
    /* Disable UART RDA/THRE/Time-out interrupt */
    UART_DisableInt(UART1, UART_INTEN_WKIEN_Msk);
    g_bWait = TRUE;

    printf("\n\n%x\n",UART_READ(UART1));
    printf("\n\n%x\n",UART_READ(UART1));
    printf("\n\n%x\n",UART_READ(UART1));
    printf("\n\n%x\n",UART_READ(UART1));

    printf("\n\nUART Wakeup Compensation Sample Demo End.\n");
}
```

3 軟體與硬體環境

- **軟體環境**
  - BSP 版本
    - M480 Series BSP CMSIS V3.04.000
  - IDE 版本
    - Keil uVersion 5.26
- **硬體環境**
  - 電路元件
    - NuMaker-PFM-M487 or other M480 Development Board
  - 示意圖

    從UART1_RX(PA.2)輸入訊號，當資料達到門檻值時會喚醒M480。



4 目錄資訊

- EC_M480_UART_Wakeup_Compensation_V1.00
  - Library — Sample code header and source files
    - CMSIS — Cortex® Microcontroller Software Interface Standard (CMSIS) by Arm® Corp.
    - Device — CMSIS compliant device header file
    - StdDriver — All peripheral driver header and source files
  - SampleCode
    - ExampleCode — Source file of example code

5 如何執行範例程式

1. 根據目錄資訊章節進入 ExampleCode 路徑中的 KEIL 資料夾，雙擊 UART_Wakeup_Compensation.uvproj
2. 進入編譯模式介面
   a. 編譯
   b. 下載代碼至記憶體
   c. 進入 / 離開除錯模式
3. 進入除錯模式介面
   a. 執行代碼

6 修訂紀錄

Date	Revision	Description
Oct. 1, 2019	1.00	1. 初始發布.

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.